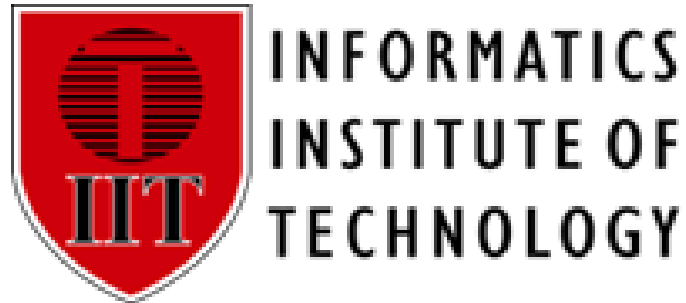




ICT A/L Seminar - 2014
57, Ramakrishna Road,
Colombo 06.

www.iit.ac.lk

01125238983 / 01125238984

Part – B Essay

Questions (1)(2)(3)(4)

(1) (a) (i)

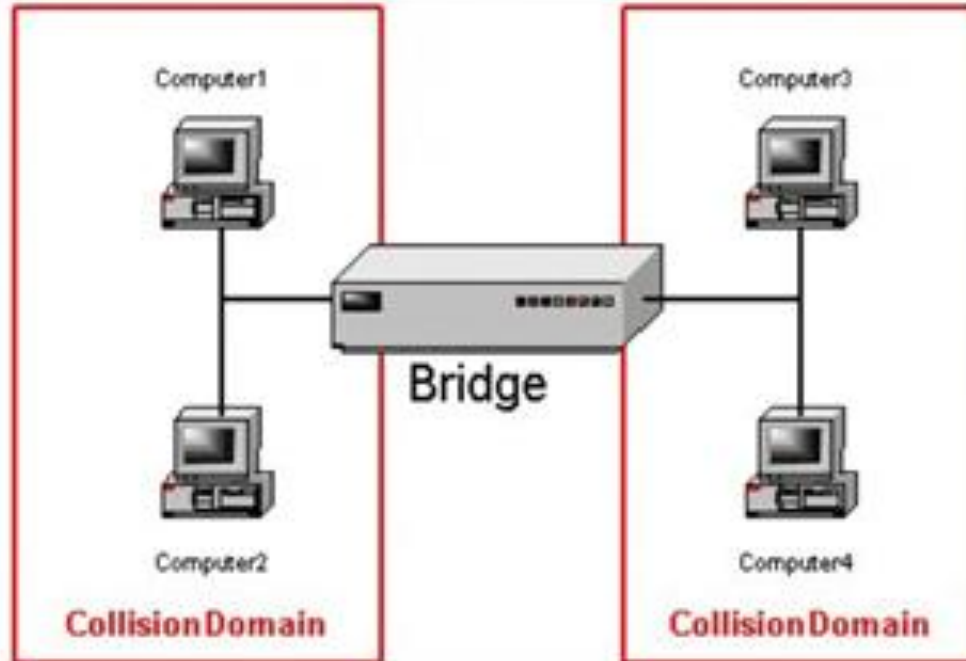
- **Repeater**

Network device used to regenerate or replicate a signal. Repeaters are used in transmission systems to regenerate analog or digital signals distorted by transmission loss.

(1)(a)(i)

- **Bridges**

In telecommunication networks, a bridge is a product that connects a local area network (LAN) to another local area network that uses the same protocol (for example, Ethernet or token ring).



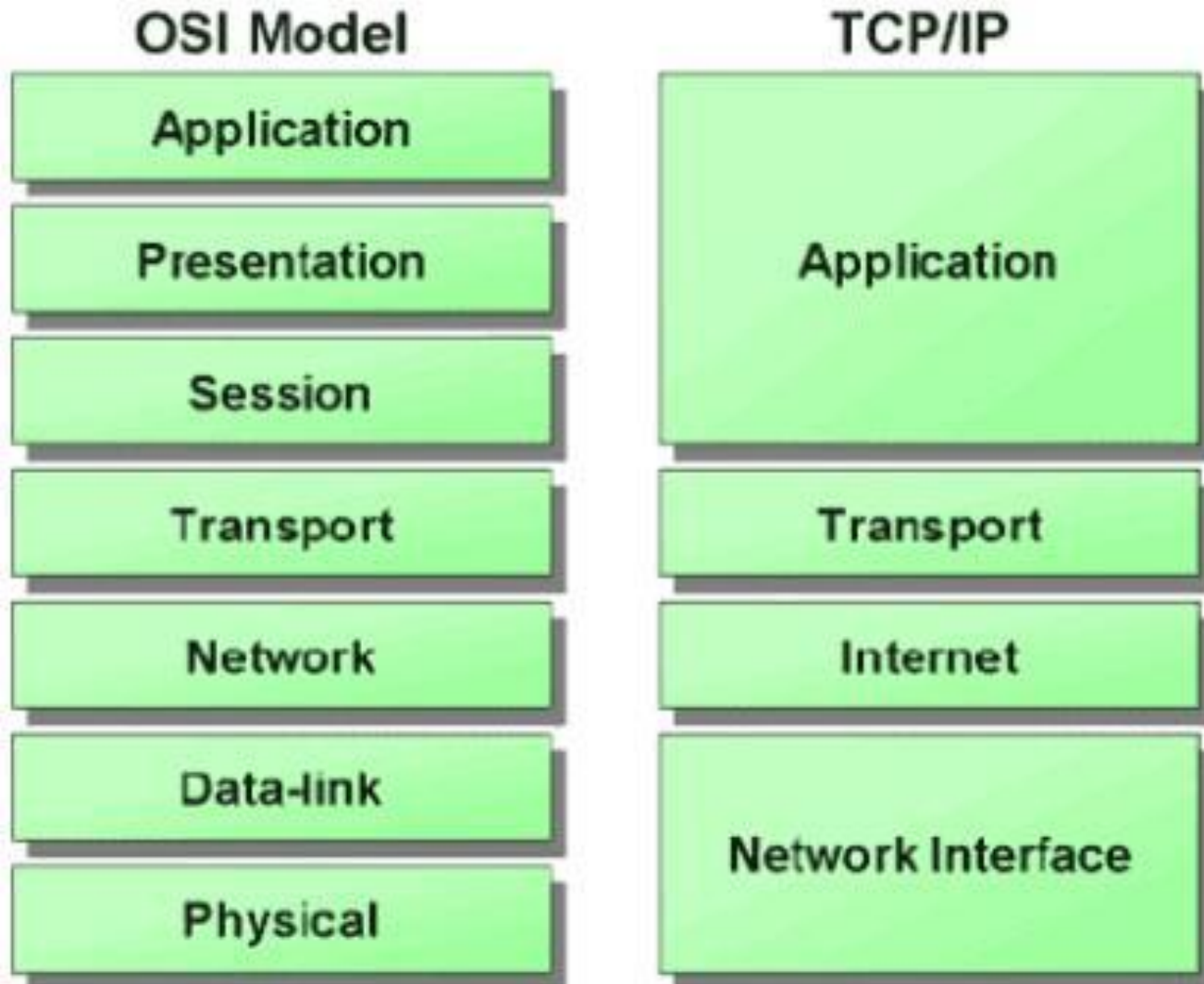
(1) (a)(i)

Router

- In packet-switched networks such as the Internet, a router is a device or, in some cases, software in a computer, that determines the next network point to which a packet should be forwarded toward its destination.
- The router is connected to at least two networks and decides which way to send each information packet based on its current understanding of the state of the networks it is connected to
- A router is often included as part of a network switch.
- A router may create or maintain a table of the available routes and their conditions and use this information along with distance and cost algorithms to determine the best route for a given packet.

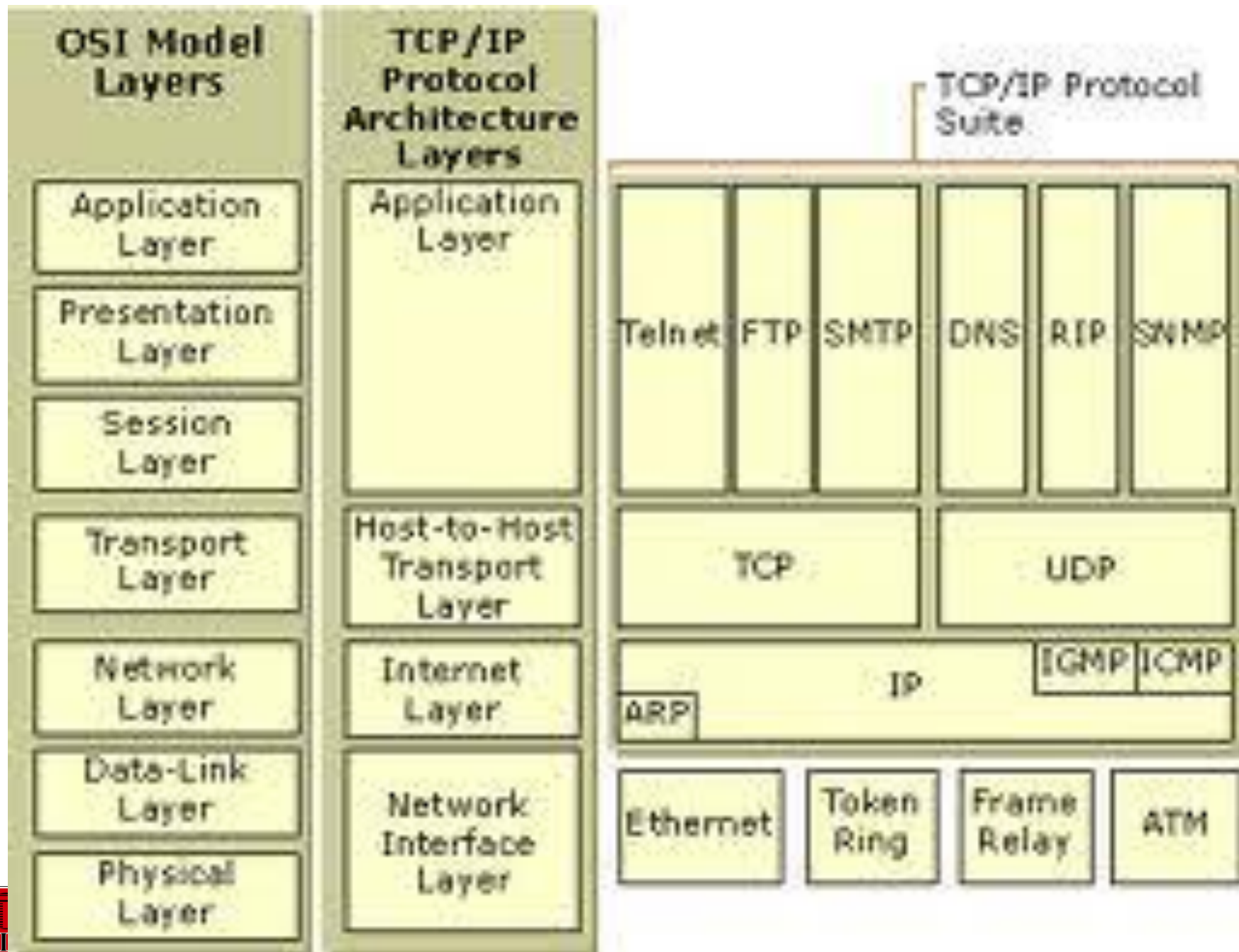


(1) (a) (ii)



TCP/IP and the OSI model

Layer #	Name	Mnemonic	Encapsulation Units	Devices or Components	Keywords/Description
7	Application	All	data	PC	Network services for application processes, such as file, print, messaging, database services
6	Presentation	People	data		Standard interface to data for the application layer. MIME encoding, data encryption, conversion, formatting, compression
5	Session	Seem	data		Interhost communication. Establishes, manages and terminates connection between applications
4	Transport	To	segments		End-to-end connections and reliability. Segmentation/desegmentation of data in proper sequence. Flow control
3	Network	Need	packets	router	Logical addressing and path determination. Routing. Reporting delivery errors
2	Data Link	Data	frames	bridge, switch, NIC	Physical addressing and access to media. Two sublayers: Logical Link Control (LLC) and Media Access Control (MAC)
1	Physical	Processing	bits	repeater, hub, transceiver	Binary transmission signals and encoding. Layout of pins, voltages, cable specifications, modulation



(1) (a) (iii)

- **Connection orientated:** Before data is transferred, a connection needs to be opened and acknowledged. Once established, data is transferred and acknowledged to ensure reliable transfer.
- **Connectionless:** Data is simply transmitted when ready (no connection establishment phase). However, data is unacknowledged and therefore unreliable.

An example of connection-oriented protocol is (TCP) and a connectionless protocol is (UDP). TCP makes a connection and checks whether the data is received, and resends if it is not. UDP is a connectionless protocol, it does not guarantee delivery by first connecting and checking whether data is received.

(1)(iv)

- The physical topology of a network is determined by the capabilities of the network access devices and media, and the cost associated with cabling or telecommunications circuits.
- The logical topology, in contrast, is the way that the signals act on the network media, or the way that the data passes through the network from one device to the next without regard to the physical interconnection of the devices. A network's logical topology is not necessarily the same as its physical topology

(1)(b)

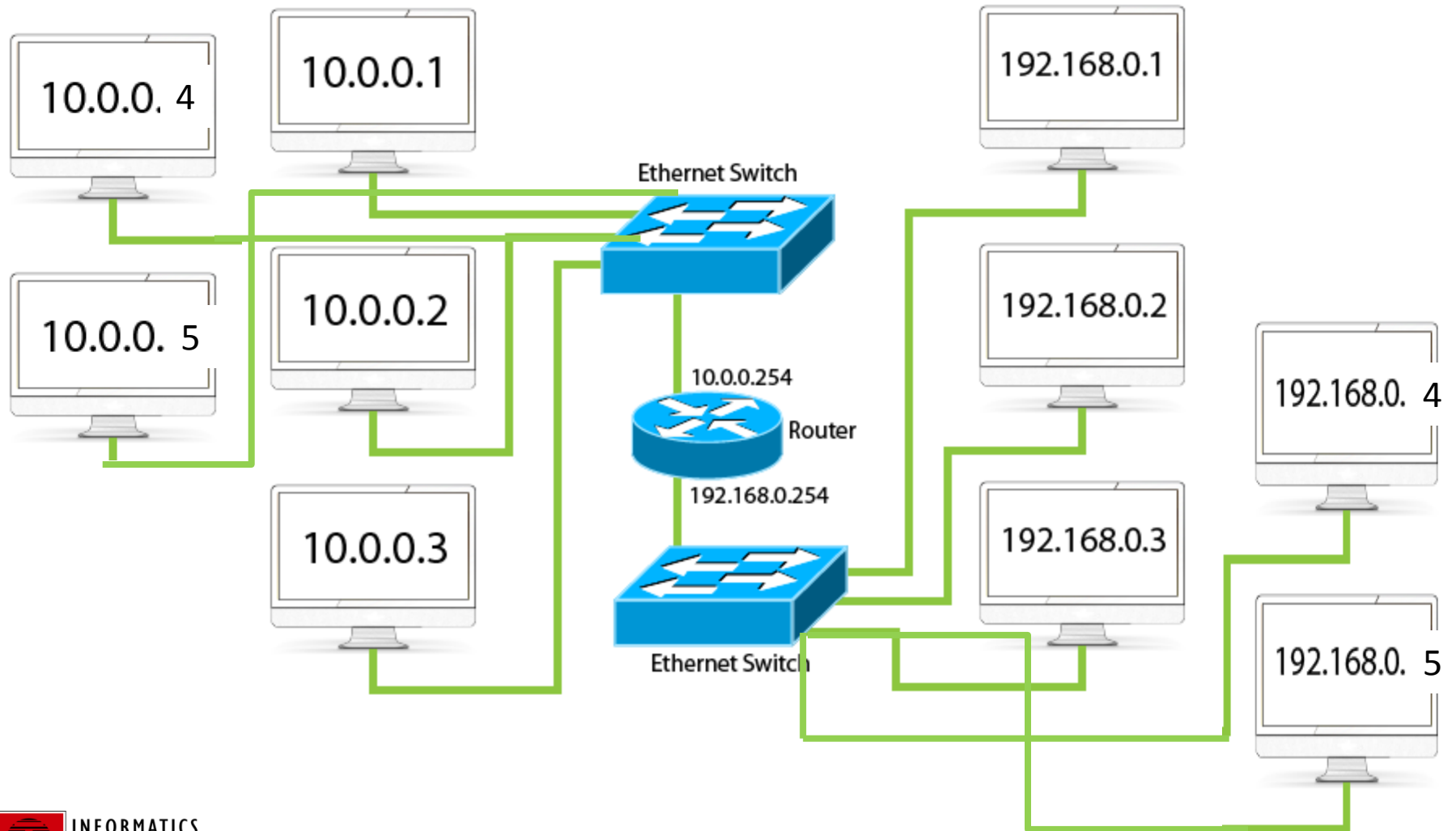
Class	Leading bits in address (binary)	Range of first octet (decimal)	Start address	End address	General allocation
A	0	0–127	0.0.0.0	127.255.255.255	Very large international organization.
B	10	128–191	128.0.0.0	191.255.255.255	Large colleges
C	110	192–223	192.0.0.0	223.255.255.255	Small and mid sized businesses
D	1110	224–239	224.0.0.0	239.255.255.255	Multicasting
E	1111	240–255	240.0.0.0	255.255.255.255	Research work

(b)

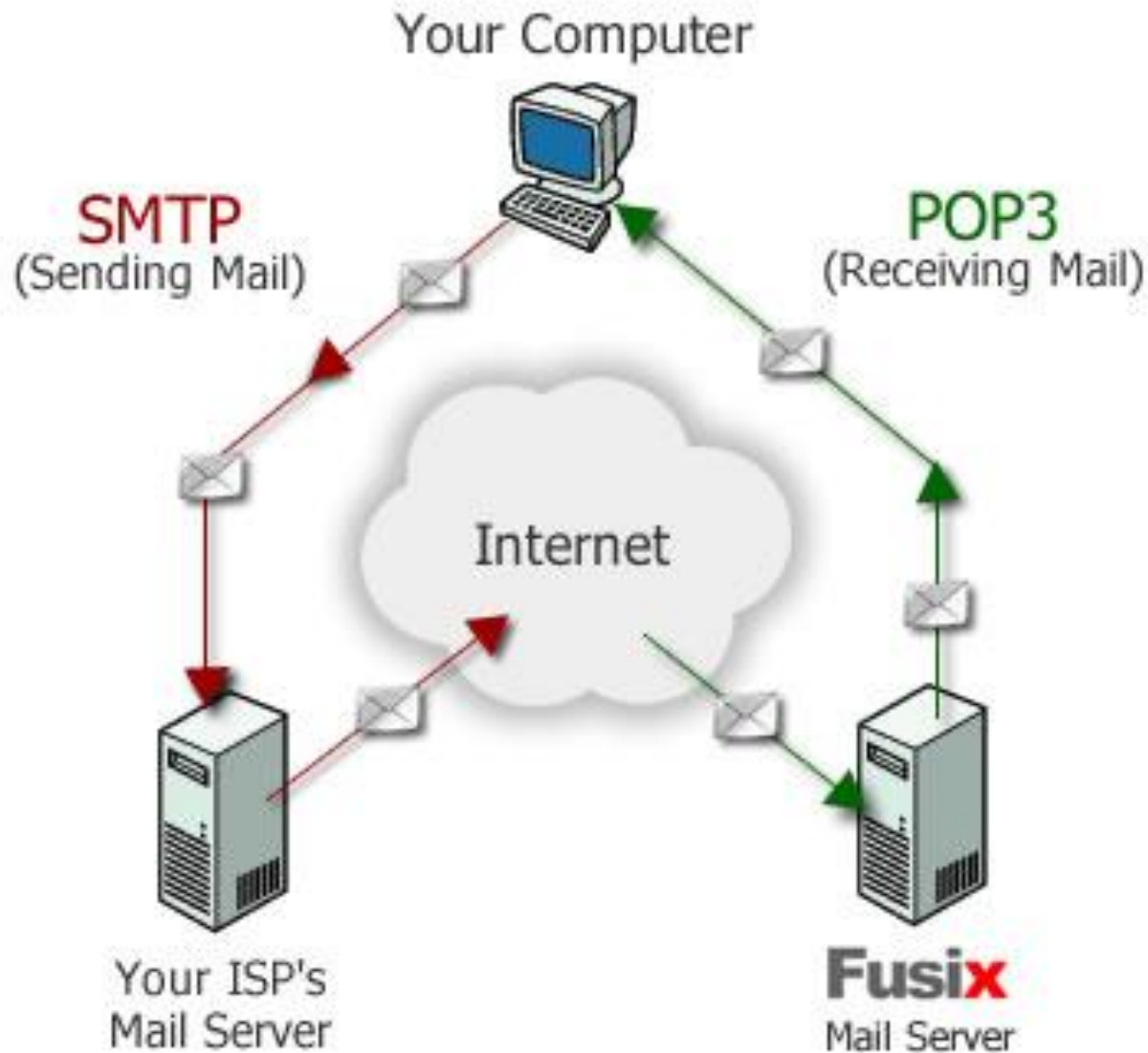
200.168.80.8/27

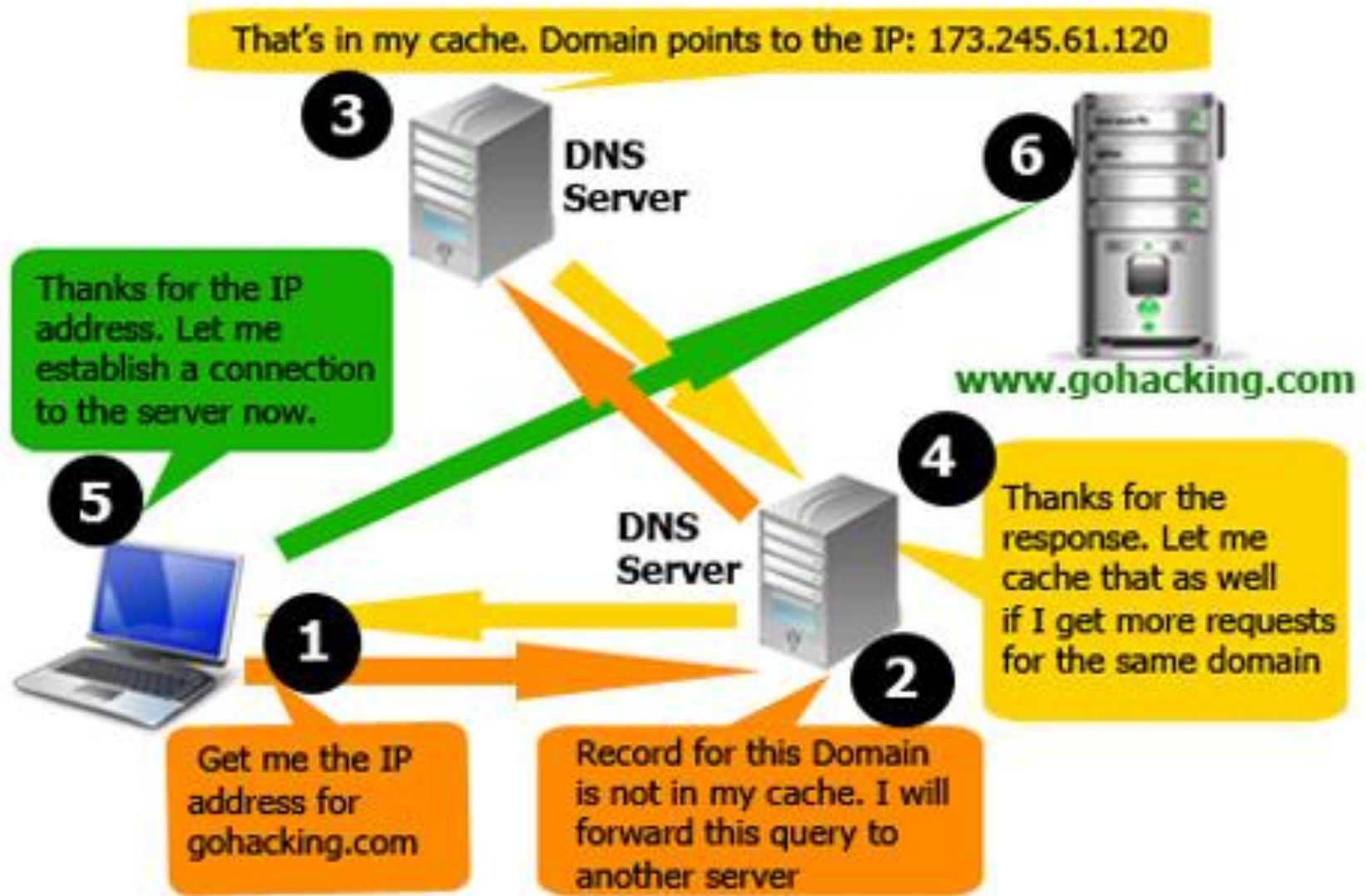
- Subnet mask : 255.255.255.224
- Number of direct hosts: 30
- network address: 200.168.80.0
- First usable host: 200.168.80.1
- Last usable host: 200.168.80.30
- Broadcast address: 200.168.80.31

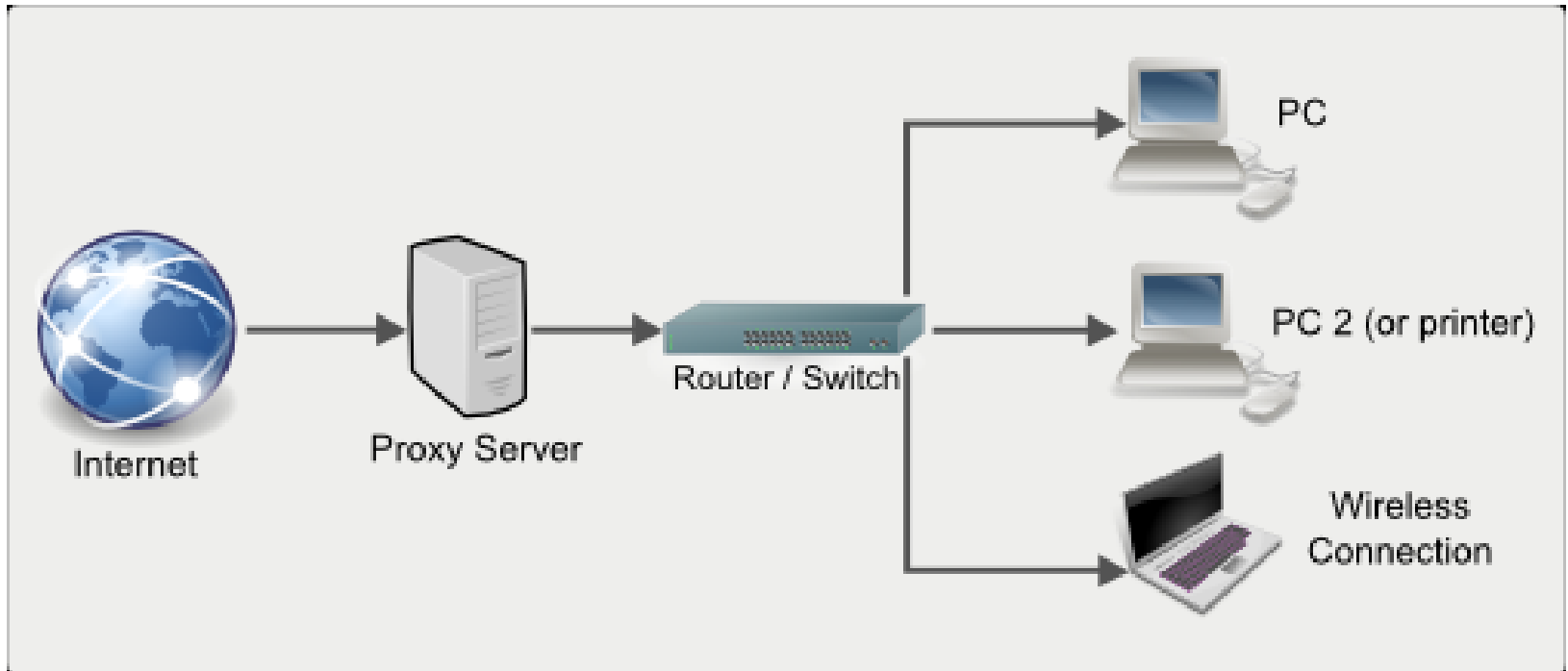
1(c) i)



1(c) ii







Difference between IP address and mac address

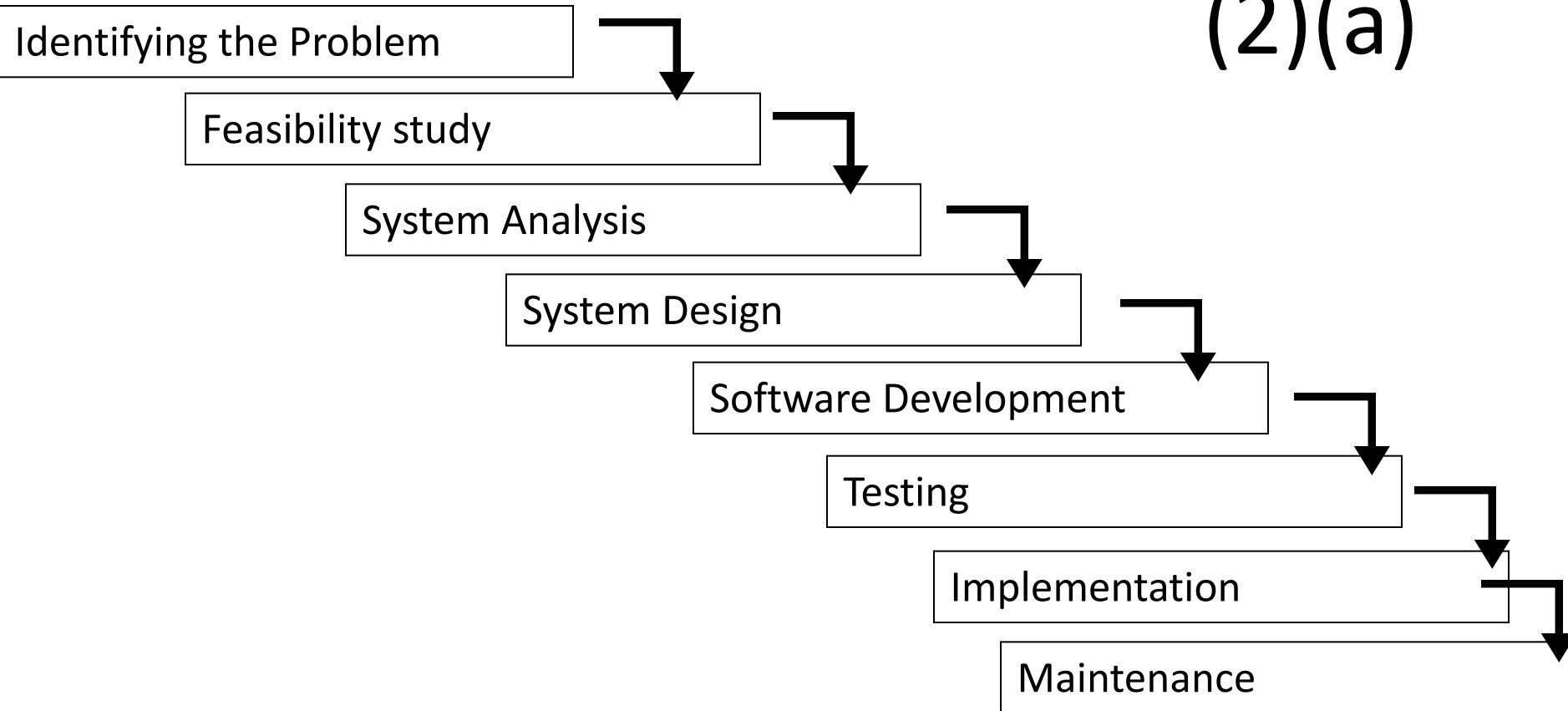
- MAC address is supposedly unique to each network interface card while an IP address is usually replaced
- An IP address reveals which element on which network it is while the same cannot be extracted from a MAC address.
- An IP address is usually assigned by the network administrator or internet service provider, you are either provided a static one at the beginning or given a dynamic one every time you connect to the network.
- This is not true with MAC addresses as it is already embedded on the device or the network card during manufacturing. It is supposed to be permanent and could not be changed by anyone as it was meant to identify a specific network interface card no matter where it is in the world.

Useful networking commands

- **IPCONFIG** - Ipconfig is a MS-DOS utility that can be used from MS-DOS and a MS-DOS shell to display the network settings currently assigned and given by a network. This command can be utilized to verify a network connection as well as to verify your network settings.
- **PING** - Ping is one of the most commonly used and known commands. Ping allows a user to ping another network IP address. This can help determine if the network is able to communicate with the network.
- **NSLOOKUP** - The nslookup MS-DOS utility that enables a user to do a [reverse lookup](#) on an IP address of a domain or host on a network.

- **HOSTNAME** - The hostname command displays the host name of the Windows XP computer currently logged into.
- **NETSTAT** -The netstat command is used to display the TCP/IP network protocol statistics and information.
- **WINIPCFG** - The winipcfg command available in Windows allows a user to display network and network adapter information. Here, a user can find such information as an IP address, Subnet Mask, Gateway, etc...
- **PATHPING**-Pathping is a MS-DOS utility available for Microsoft Windows 2000 and Windows XP users. This utility enables a user to find network latency and network loss.

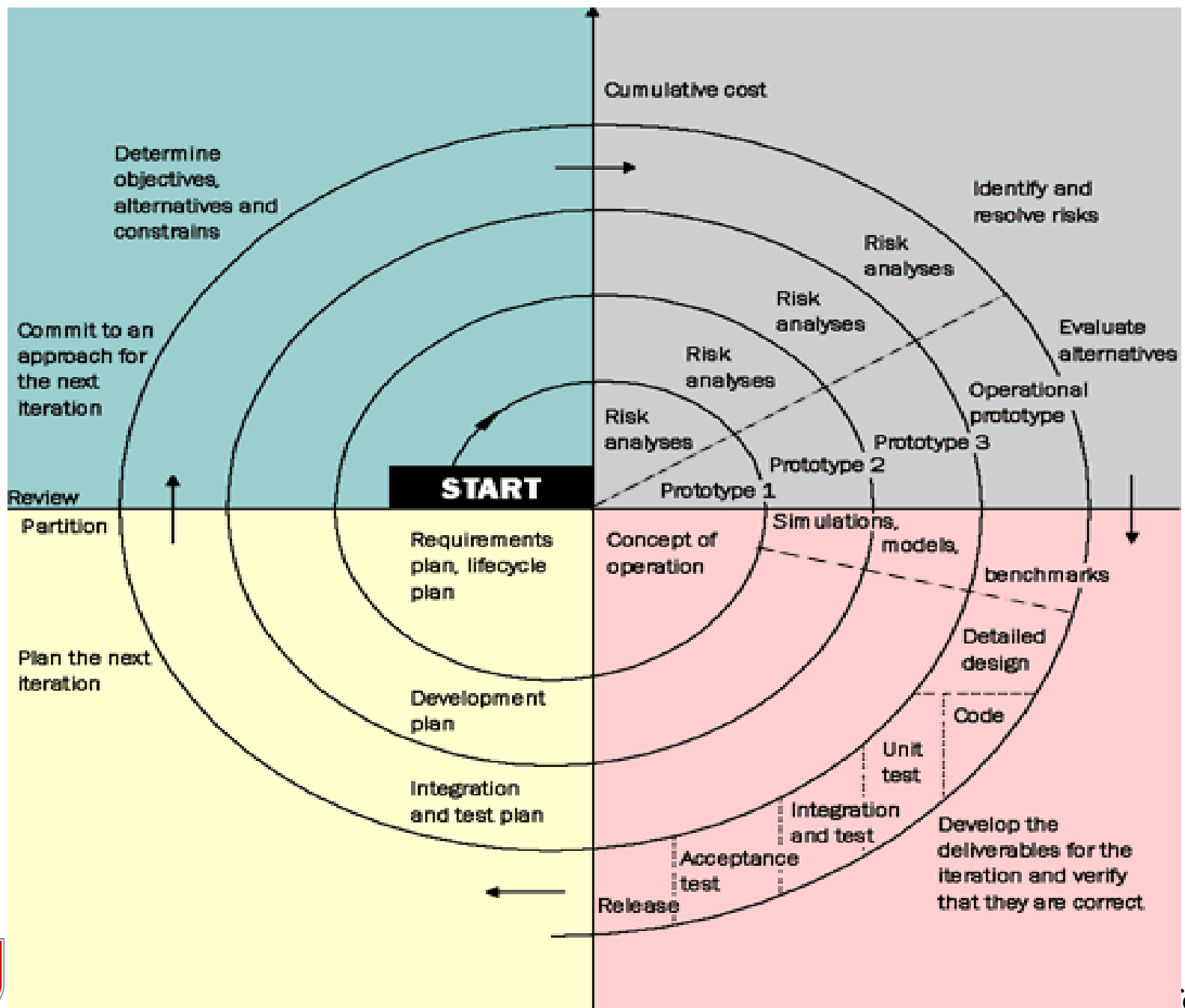
(2)(a)



Waterfall model is simple to implement and also the amount of resources required for it are minimal therefore it suits for simple systems which the requirements are clearly understood.

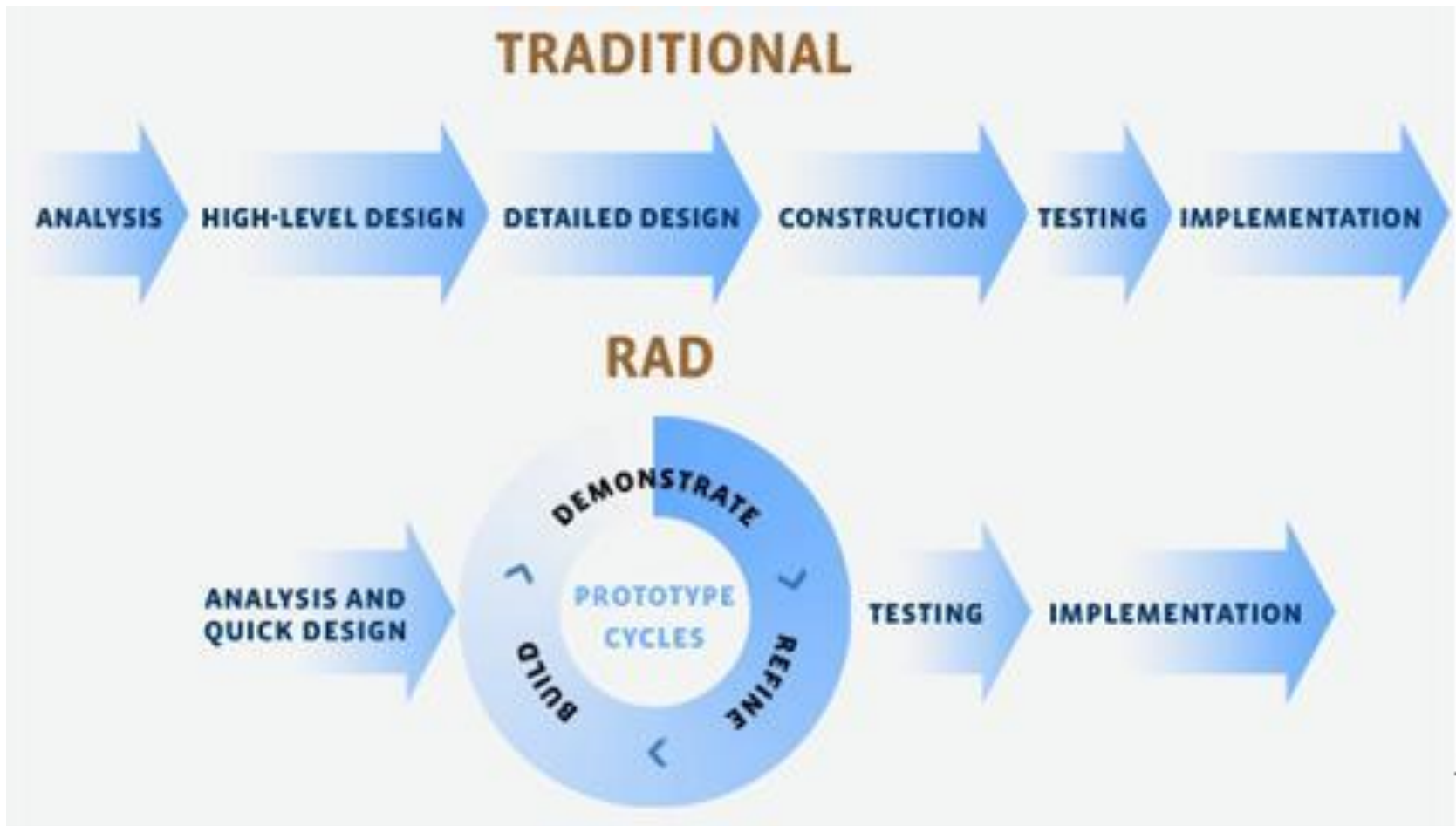
(4)(a)(i)

Identifying	Identifying the problem and doing an initial investigation about it.
Feasibility	Checking whether the project is economically, technically and operationally feasible.
Analyzing	Conducting a detailed requirement identification and make requirement specification.
Designing	Planning the new system and create the design specification
Coding	Convert the design spec. into working programmes .
Testing	Checking the system to identify all types of bugs.
Implementation	Transferring the system to user's environment. Fixing new equipment , Training Users
Maintenance	Changing according to the new requirements

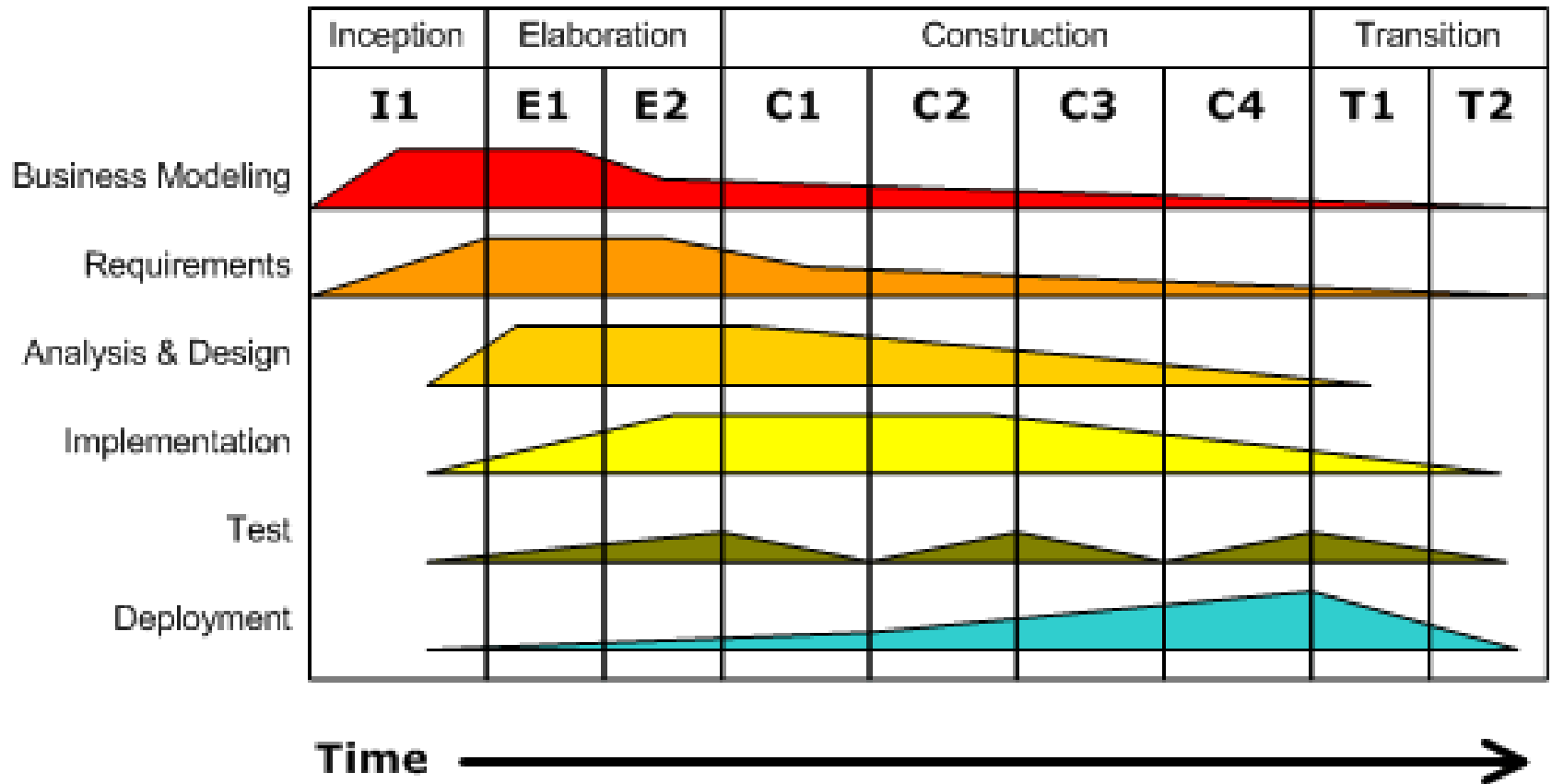


- The spiral model is intended for large, expensive and complicated projects which risks can be also there.

Rapid Application Development (RAD)



- The lack of extensive pre-planning generally allows software to be written much faster, and makes it easier to change requirements therefore it is suitable for projects which has tight schedules and deadlines.



- RUP is not a single concrete prescriptive process, but rather an adaptable process framework, intended to be tailored by the development organizations and software project teams that will select the elements of the process that are appropriate for their needs. RUP is a specific implementation of the Unified Process.
- It is suitable for large and complex systems.

(2)(b)

- Requirement analysis
 - Requirement elicitation – gather information about the current system using fact finding method.
 - Analyzing the gathered requirements.
 - Verifying the requirements
 - Identify functional and non functional requirements
 - Create System requirements specification doc.

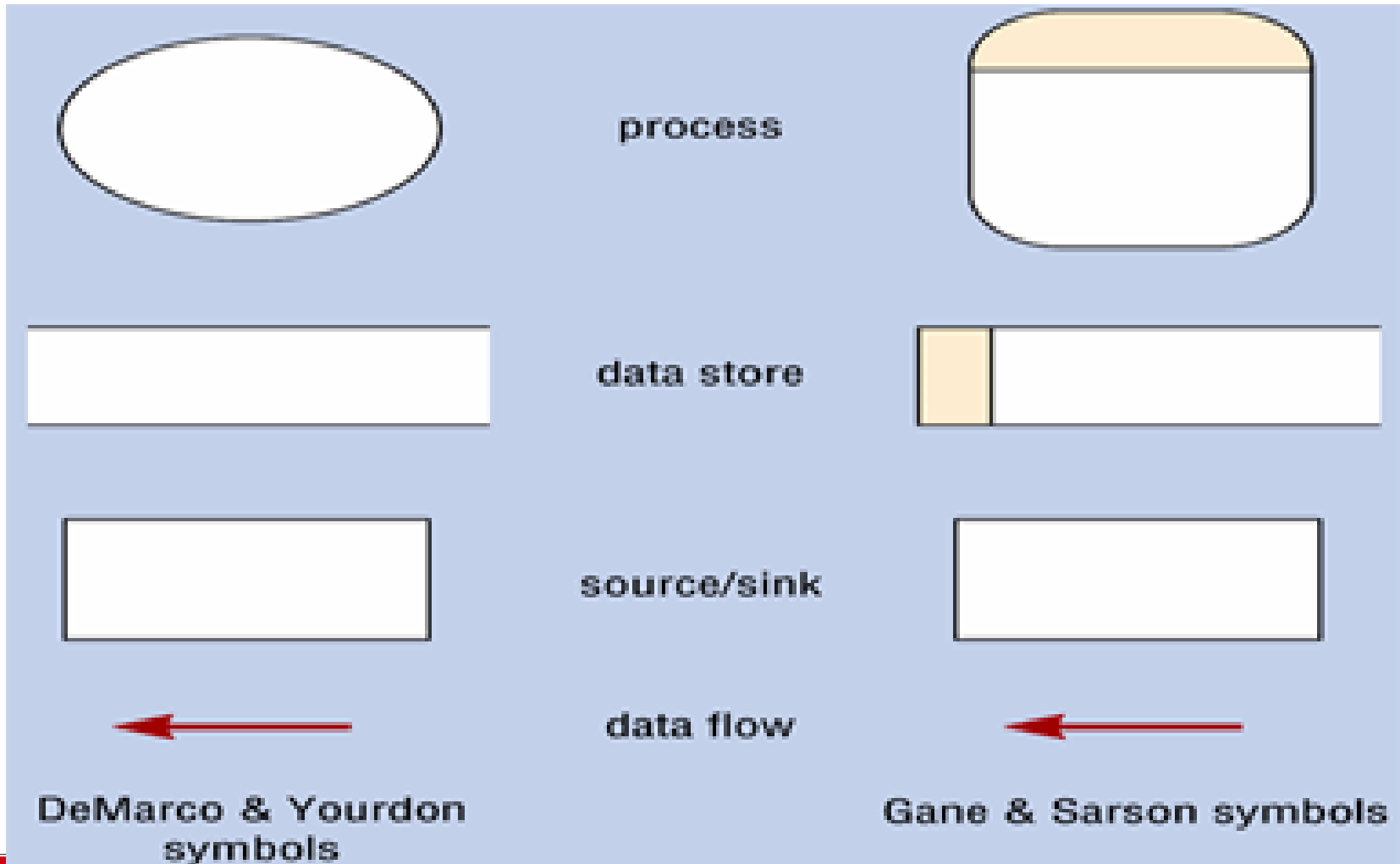
(2)(b)(ii)

- Functional requirements:
 - Members who registered newly should be displayed as a table.
 - List of members who didn't pay the membership fee should be displayed.
 - Librarian should be able to make reports about the book categories from the system.
 - An online feature to view the new arrivals for a member

(2)(b)(ii)

- Non-Functional requirements:
 - The interfaces in the system should be user friendly.
 - The library staff should be able to learn the system quickly.
 - The system interfaces should be loaded quickly.
 - Library staff has to access the system as administrators while members has to log to the system as systems.

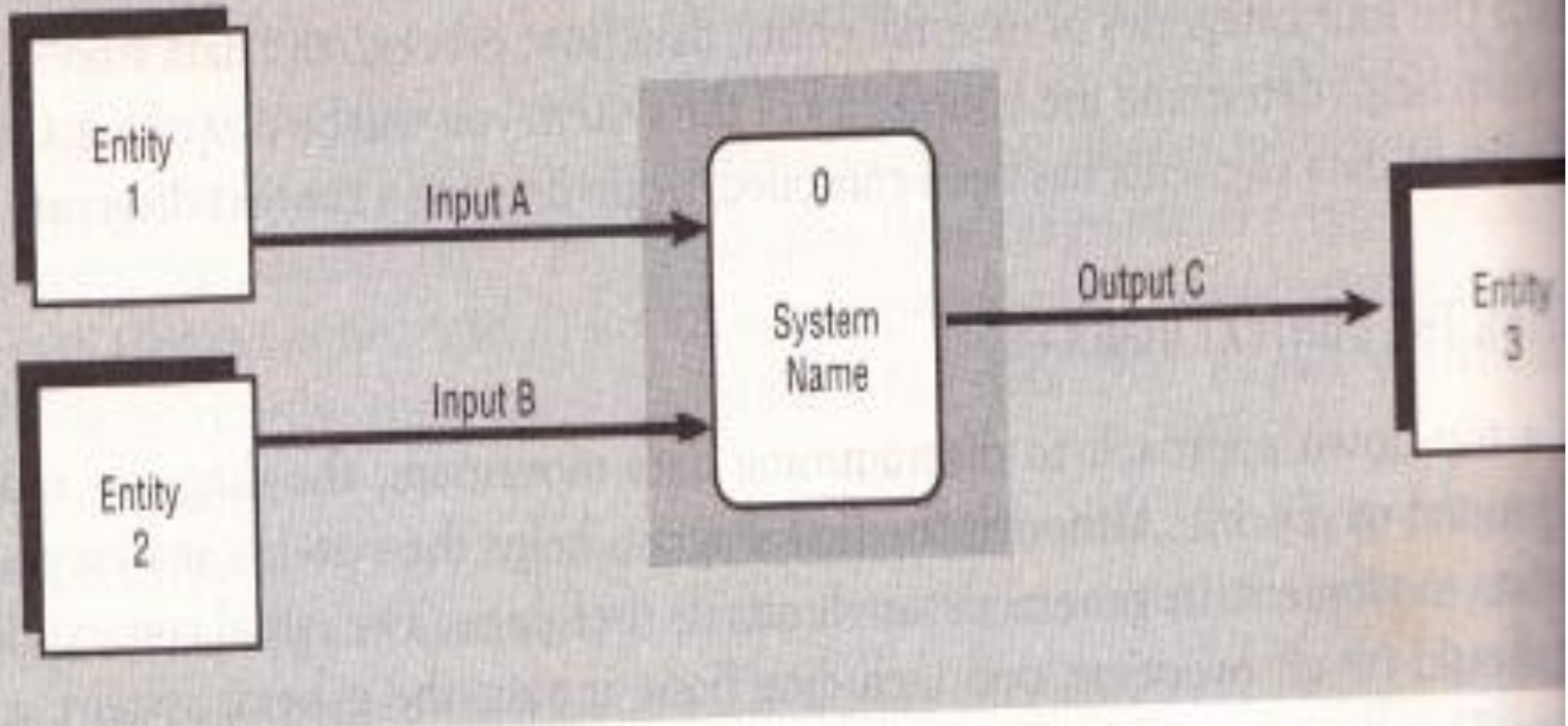
DFD symbols

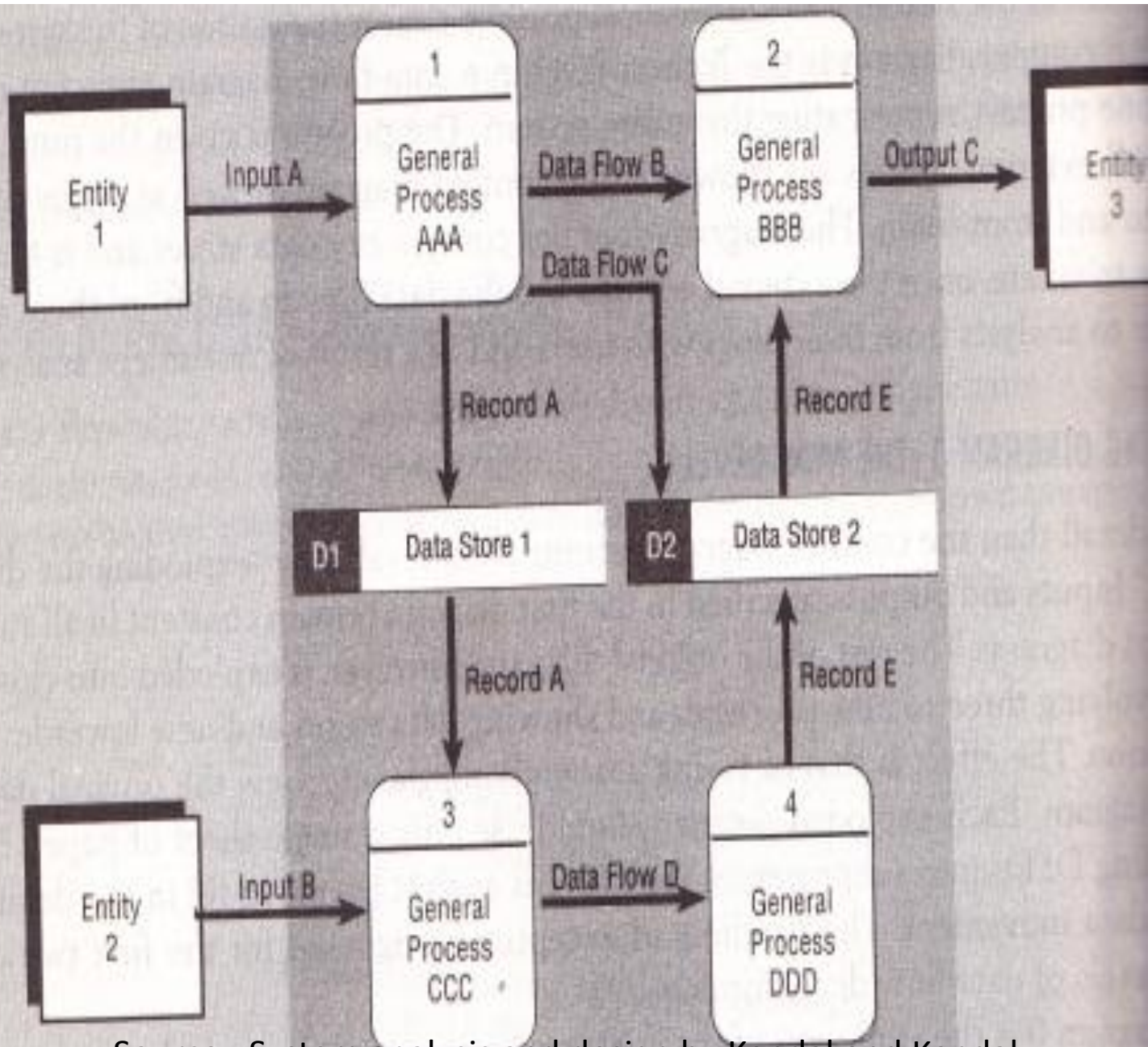


Importance of DFD

- It helps to organize the data flow properly - Programmers use DFDs to plan exactly how their new program is going to accomplish its intended purpose.
- Decision building – DFDs help the programmer figure out what options the programs will need in order to handle the data it is given.
- Presentation – It helps to present the software plan to a group of stake holders despite of their IT literacy.
- Adaptability – If the programmer requires to change the plan .

(2)(c)(i)





Source : System analysis and design by Kendal and Kendal

(2)(c)(ii) Levels of testing

- Unit testing
- Integration testing
- System testing
- Acceptance testing

Unit testing

- Unit testing, also known as component testing, refers to tests that checks the accurate functionality of a specific section of code.
- The programme codes / units/ modules/ are individually tested.

Integration testing

- Integration testing works to expose defects in the interfaces and interaction between integrated components (modules).
- The individual components are integrated to see whether the functionality of the software.
- Interdependency of the modules will be tested.

Systems testing

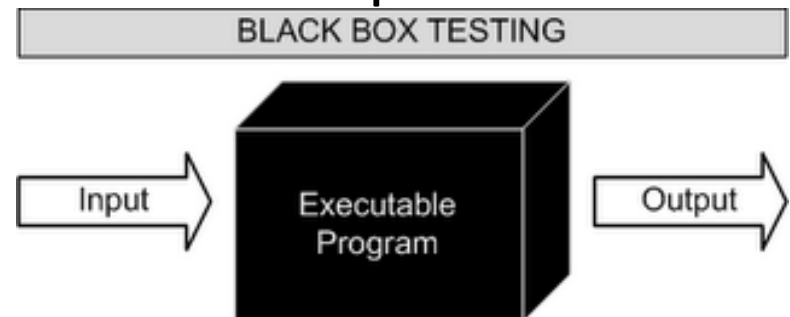
- The complete system is tested after integrating the all the components.
- It evaluates the system's compliance with its specified requirements.
- Most of the this falls into the category of black box testing.
- It includes many types of testing methods such as Graphical user interface testing, Usability testing ,Performance testing ,Compatibility testing , Error handling testing ,Load testing, Volume testing, Stress testing ,Security testing, Scalability testing , Sanity testing , Smoke testing , Regression testing, Reliability testing, Installation testing, Maintenance testing etc..

Acceptance testing

- **Acceptance Testing** is a level of the software testing process where a system is tested for acceptability.
- The purpose of this test is to evaluate the system's compliance with the business requirements and assess whether it is acceptable for delivery.
- Acceptance testing can mean one of two things:
 - A smoke test is used as an acceptance test prior to introducing a new build to the main testing process, i.e. before integration or regression.
 - Acceptance testing performed by the customer, often in their lab environment on their own hardware, is known as user acceptance testing (UAT).

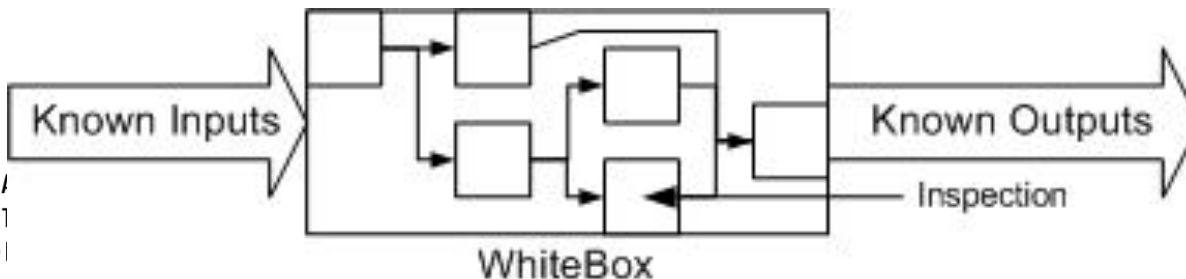
(2)(d)(iii)Black box testing

- Also known as *functional testing*. Software testing technique whereby the internal workings of the item being tested are not known by the tester. For example, in a black box test on a software design the tester only knows the inputs and what the expected outcomes should be and not how the program arrives at those outputs.
- The tester does not ever examine the programming code and does not need any further knowledge of the program other than its specifications.
- The input is tested against the intended output



(2)(d)(iii) White box testing

- Also known as *glass box*, *structural*, *clear box* and *open box testing*. A software testing technique which the tester has to know about the internal workings of the programme and how the test data functions.
- It uses specific knowledge of programming code to examine outputs. The test is accurate only if the tester knows what the program is supposed to do.
- When a certain input is given until the output is obtained all the operations need to be recognized and if there are errors they must be detected



(2)(d)(iv)

- Identify a function – Taking a photo in in “night mode”
- Plan the test – take the camera and set it to night mode.
- Decide the output – You can download a picture or check the user manual to get an idea of the picture quality.
- Execute – Take a photograph in outside when it is dark.
- Compare the result with expected output.

(3)(a)

- Heat (B) $\rightarrow$ Temperature is less than or equal to 24°C (0) / otherwise (1)
- Humidity (C) $\rightarrow$ Low humidity (0) / High humidity (1)
- Presence (D) $\rightarrow$ If no one in the room (0) / If any one in the room (1)
- Based on the following conditions the A/C will be switched off
- Any time user can manually switch off the A/C (0)
- At least two of the sensors should get negative (0) input.
- Consider switching off process of the A/C as Boolean 1 and switching on process as Boolean 0 and answer the following conditions.

(3)(a)

A	B	C	D	F(A,B,C,D)	Minterms
0	0	0	0	1	ABCD
0	0	0	1	1	
0	0	1	0	1	
0	0	1	1	1	
0	1	0	0	1	
0	1	0	1	1	
0	1	1	0	1	
0	1	1	1	1	
1	0	0	0	1	
1	0	0	1	1	
1	0	1	0	1	
1	0	1	1	0	
1	1	0	0	1	
1	1	0	1	0	
1	1	1	0	0	
1	1	1	1	0	

A	B	C	D		Minterm
0	0	0	0	1	$\bar{A} \cdot \bar{B} \cdot \bar{C} \cdot \bar{D}$
0	0	0	1	1	$\bar{A} \cdot \bar{B} \cdot \bar{C} \cdot D$
0	0	1	0	1	$\bar{A} \cdot \bar{B} \cdot C \cdot \bar{D}$
0	0	1	1	1	$\bar{A} \cdot \bar{B} \cdot C \cdot D$
0	1	0	0	1	$\bar{A} \cdot B \cdot \bar{C} \cdot \bar{D}$
0	1	0	1	1	$\bar{A} \cdot B \cdot \bar{C} \cdot D$
0	1	1	0	1	$\bar{A} \cdot B \cdot C \cdot \bar{D}$
0	1	1	1	1	$\bar{A} \cdot B \cdot C \cdot D$
1	0	0	0	1	$A \cdot \bar{B} \cdot \bar{C} \cdot \bar{D}$
1	0	0	1	1	$A \cdot \bar{B} \cdot \bar{C} \cdot D$
1	0	1	0	1	$A \cdot \bar{B} \cdot C \cdot \bar{D}$
1	0	1	1	0	—
1	1	0	0	1	$A \cdot B \cdot \bar{C} \cdot \bar{D}$
1	1	0	1	0	—
1	1	1	0	0	—
1	1	1	1	0	—



Standard SOP =

$\Sigma (0,1,2,3,4,5,6,7,8,9,10,13,14,15)$

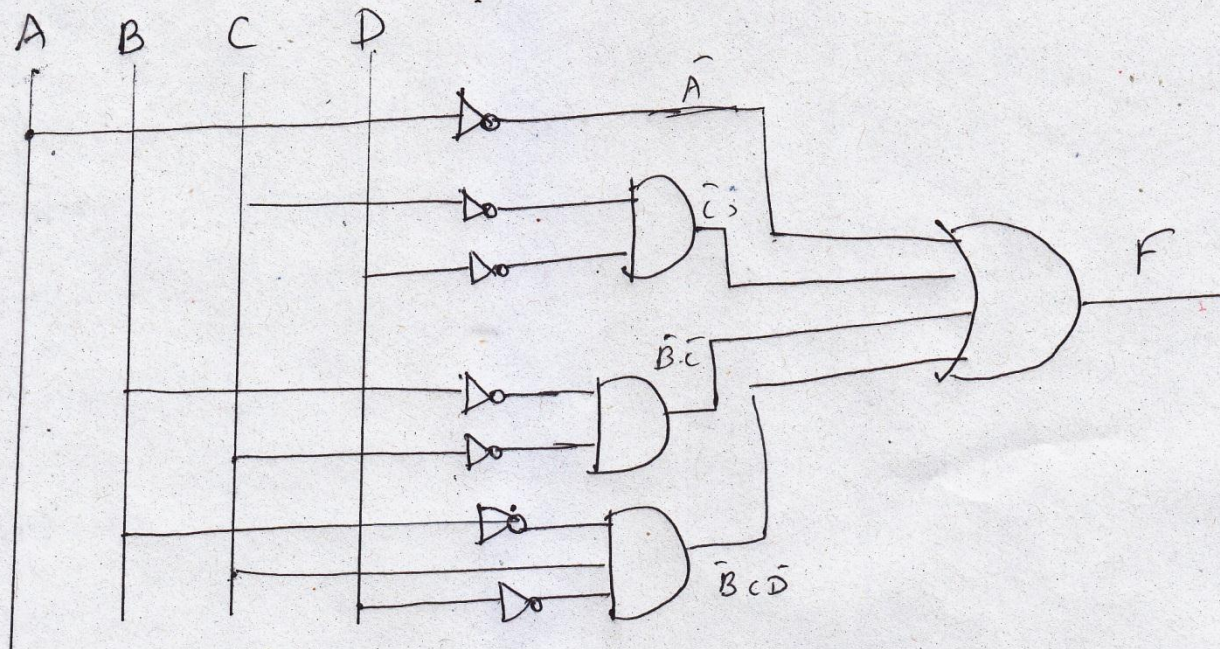
$$\begin{aligned} &= \bar{A}\bar{B}\bar{C}D + \bar{A}\bar{B}\bar{C}\bar{D} + \bar{A}\bar{B}C\bar{D} + \bar{A}\bar{B}CD + \bar{A}B\bar{C}\bar{D} + \bar{A}B\bar{C}D + \\ &\quad \bar{A}B\bar{C}\bar{D} + \bar{A}B\bar{C}D + A\bar{B}\bar{C}\bar{D} + A\bar{B}\bar{C}D + A\bar{B}C\bar{D} + A\bar{B}C\bar{D} \end{aligned}$$

(1)(iii)

		CD			
		00	01	11	10
AB	00	1	1	1	1
	01	1	1	1	1
	11	1	0	0	0
	10	1	1	1	0

Simplified expressions:

$$\bar{A} + \bar{C}\bar{D} + \bar{B}\bar{C} + \bar{B}C\bar{D} //$$



(b) (i)

- 0000000000000000 – 1111111111111111
- 0 to $2^{16} - 1$

(b) (ii)

- 0101111000110001

Virtual memory address space in bits = 2^{16}

Virtual memory address space

in kilobytes

$$= 2^{16} / 2^3 * 2^{10} = 2^3$$

$$= 2^3 = 8\text{KB}$$

(b) (iii)

- 0101111000110001

If page size is 512bytes

$$\text{Bits} = 512 * 8 * = 2^9 * 2^3 = 2^{12} \text{ bits}$$

If 12 bits are for a page size, No of bits are:

$$= 2^{16} / 2^{12} = 2^4 = 16 \text{ pages}$$

(b) (iii)

Page size = 12 bits

0101 111000110001



Page number
4 bits



Offset 12 bits

Q4 (a) i) ii) iii)

- `<input>` Allows input controls to be added to a form.
 - Attribute name
- `<ol>` Starts a Ordered list
 - Attribute type
- `<frame>` Allows browser window to be divided in to different sections called frames.
 - Attribute src

Q4 (b) i)

```
<center>  
  <a href="scenarios.html" target="_blank">  
      
  </a>  
</center>
```

Q4 (b) ii)

<dl>

<dt>Importance </dt>

<dd>Famous because of the ancient frescoes </dd>

<dt> Nick name </dt>

<dd> The Lion's Rock </dd>

</dl>

Q4 (b) iii)

```
<link href="file1.css" rel="stylesheet" type="text/css" />
```

- This will add the CSS code in the file file1.css to the web page. The web page contents will be displayed according to the code in the file1.css

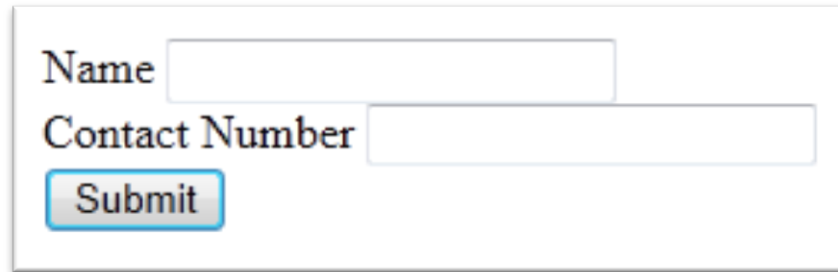
Q4 (b) iv)

p{

```
color: #000099;  
font-family: Arial, Helvetica, sans-serif;  
text-align: justify ;
```

}

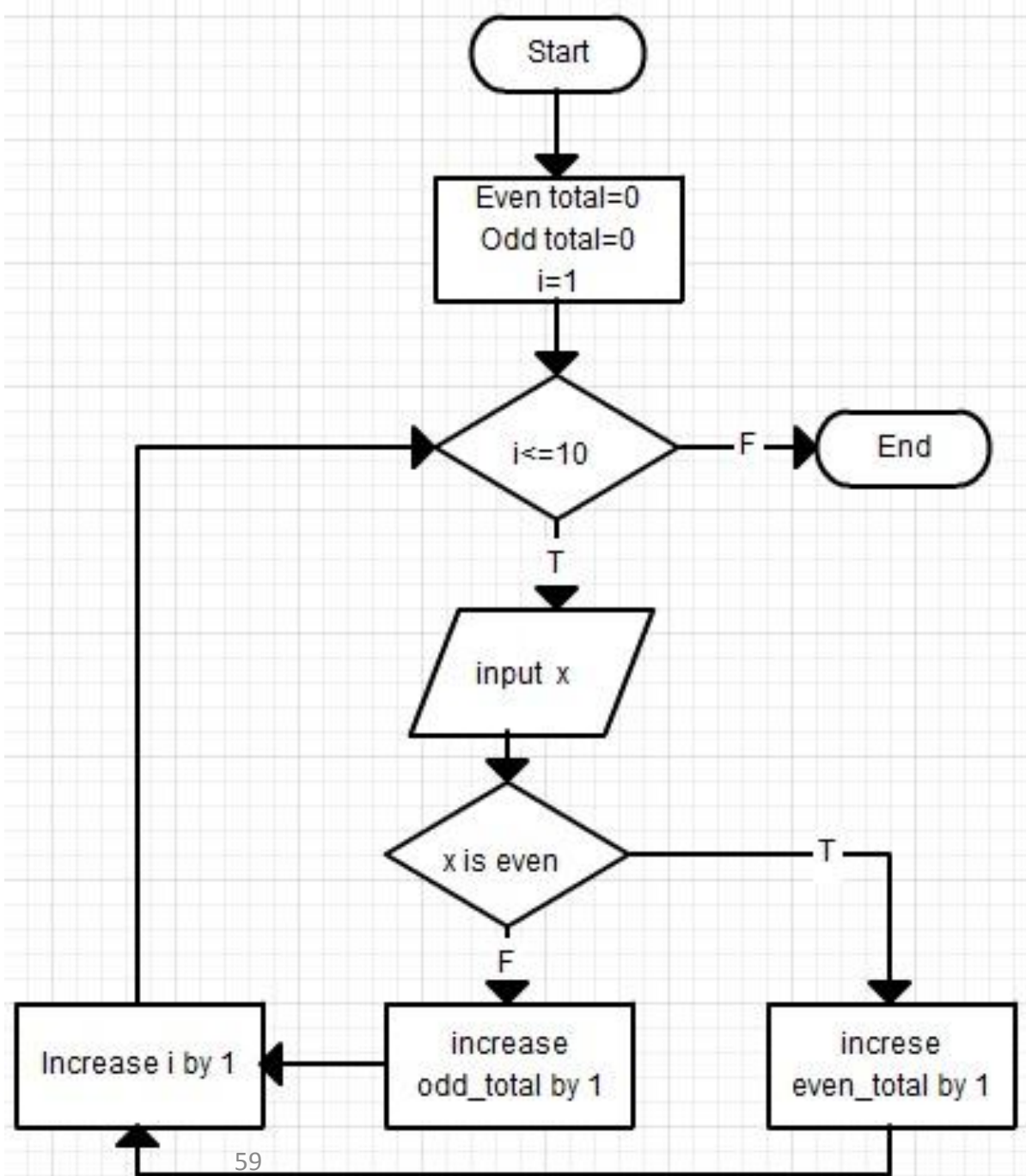
Q4 (c)



A screenshot of a web form. It contains two text input fields. The first field is labeled 'Name' and the second is labeled 'Contact Number'. Below these fields is a button labeled 'Submit'.

```
<form id="form1" name="form1" method="post" action="">  
Name      <input type="text" name="name" id="name" />  
<br />  
Contact   Number <input type="text" name="address" id="address" />  
<br />  
<input type="submit" name="submit" id="submit" value="Submit" /  
onclick="update.php">  
</form>
```

Q5 (a) i)



Q5 (a) ii)

```
even_total=0
```

```
odd_total=0
```

```
for i in range(1,11):
```

```
    x=int(input("Enter Number:"))
```

```
    if x%2==0:
```

```
        even_total=even_total+x
```

```
    else:
```

```
        odd_total=odd_total+x
```

```
Print( "Total of Odd Numbers:",odd_total)
```

```
print ("Total of Even Numbers:",even_total)
```

Q6 (b) i)

Given Code

```
y=int(input( ))  
for x in range(1,y):  
    prime=True  
    for i in range(2,x):  
        if x%i=0:  
            prime=False  
    if prime:  
        print (x)
```

Q6 (b)i)

Correct Code

```
y=input("Enter Number:")
for x in range(1,y+1):
    prime=True
    for i in range(2,x):

        if x%i==0:

            prime=False
if prime:
    print (x)
```

Q6 (b)ii)Code Corrections

```
student = {'name':'saman','no':1425,'class':'12B'}  
print student{'name'}  
print student{}  
print student.keys{}  
print student.values{}
```

```
student = {'name':'saman','no':1425,'class':'12B'}  
print (student['name'])  
print (student)  
print (student.keys())  
print (student.values())
```

Q6 (c)i)Output of Code

Q5

```
for x in range(1,11,1):  
    print "Values of x=",x  
    if x%3!=0:  
        continue  
    print "TRUE"
```

Output

Values of x= 1
Values of x= 2
Values of x= 3
TRUE
Values of x= 4
Values of x= 5
Values of x= 6
TRUE
Values of x= 7
Values of x= 8
Values of x= 9
TRUE
Values of x= 10

Q6 (c)i)

```
1. f1=open("text.txt","r")
2. line=f1.readline()
3. data=line.strip().split(",")
4. datacount=len(data)
5. for i in range(datacount-1):
6.     for k in range(i+1,datacount):
7.         if int(data[i])<int(data[k]):
8.             data[i],data[k]=data[k],data[i]
9. f1.close()
10. print (data)
```

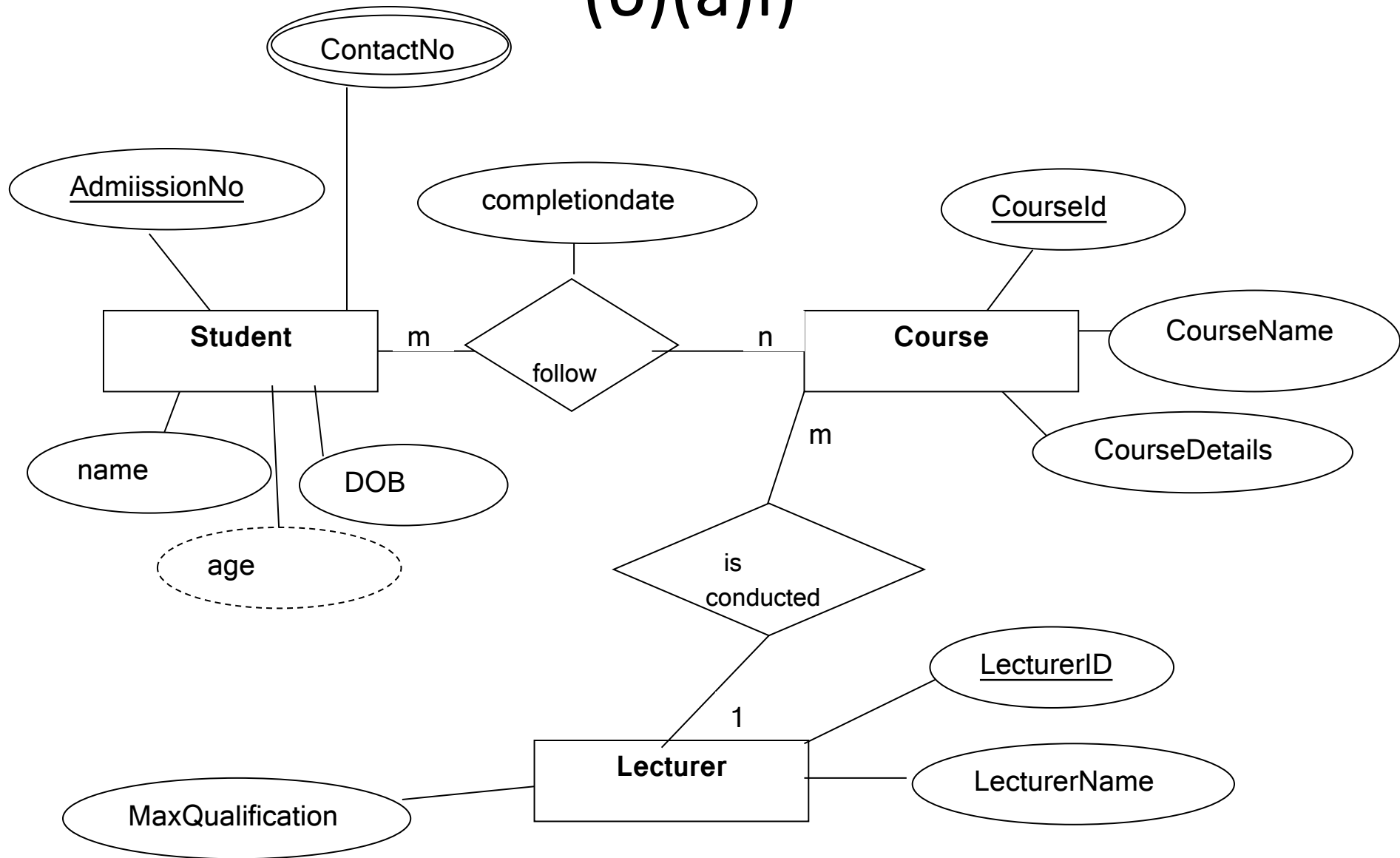
Q6 (c)i)

1. Opens the text.txt file for read only and assigns it to variable 'F1'
2. Reads the first line and assigns to the variable 'line'
3. Remove unwanted spaces in the selected section and separate the data items from commas and assigns it to the variable data the selected section becomes a list.
4. Find the length value of the 'data' and assigns it to 'datacount'
5. Perform the sorting and arrange the number set in to descending order.
6. Close the file.
7. print the list.

Output:

```
['15', '12', '9', '8', '7', '6', '4', '3', '2', '1', '0']
```

(6)(a)i



(6)(a)(ii)

Entities/Relationships	Cardinality	Reason
Student and Course	Many to many	One student can do many courses, In a course there are many students.
Lecturer and Course	One to many	One lecturer can conduct many courses. A course can be conducted by one lecturer.

(6)(a)iii)

- No

(6) (b)

- Student(AdmissionNo, Name, dob, Age)
- Course(CoursreId, CourseName, CourseDetails, LecturerId)
- ContactNo(ContactID,AdmissionNo, _ContactNo)
- Follow (AdmissionNo, CoursreId, completiondate)
- Lecturer(LecturerId, LecturerName, MaxQualifications)

(6)(c)(i)

```
create table student (AdmissionNo char(10),  
                      Name char(20),  
                      dob date,  
                      Age int  
                      primary key(Admin_no));
```

(6)(c)(ii)

```
select s.Name, f.completiondate  
from student s, follow f  
where s.AdmissionNo = f.AdmissionNo and  
f.Completiondate >=01-07-2012 and  
f.Completiondate <=20-07-2012;
```

(6)(c)(iii)

```
select distinct(l. LecturerName),  
from Lecturer l, Course c  
where l.LecturerId = c.LecturerId and  
c.LecturerId = NOT NULL;
```